International Journal of Robotics
Research and Development (IJRRD)
ISSN(P): 2250-1592; ISSN(E): 2278-9421
Vol. 11, Issue 2, Dec 2021, 39–44

DETERMINATION OF LATENCY OF AN ABB INDUSTRIAL ARM ROBOT

SAQEEF TEHNAN MANNA[1], KARTIK [2] & VINOD B[3]

[1]*Robotics Engineer, University West, Trollhattan, 461 32, Sweden*

[2]*Masters in Robotics and Automation, University West, Trollhattan, 461 32, Sweden*

[3]*Assistant Professor, Department of Mechanical Engineering, Vidyavardhaka College of Engineering, Mysore, India*

ABSTRACT

The time elapsed between deciding to take action and perceiving its consequences in a given environment is called the control latency. Commands given to robots get executed within a fixed period (deadline) i.e., Robots are controlled in real-time. In simple words, it is the time taken between action and reaction. A robot missing deadlines can result in unwanted motions and 'jerky' behavior. Latency has been a major issue in senor-robot interaction. The latency of a robot in the industry causes losses (cycle time of process increases, which increases the cost of the process). Several factories affect latency. In this paper, we compare two different codes to find out the latency in the IRB manipulator and choose the best among them.

KEYWORDS: Latency, Search function (SearchL) & Trap Routine

Received: Aug 04 2021; **Accepted**: Aug 24, 2021; **Published**: Sep 08, 2021; **Paper Id**.: IJRRDDEC20213

Original Article

1. INTRODUCTION

A robot is a machine, reprogrammable and capable of carrying out a complex series of actions automatically. Robots can be guided by an external control device, or the control may be embedded within a robot may be mobile or mounted to a base. There are different types of robots, based on their usage, functionality, and place where they are used. Different robots are used for different purposes [3].

1.1 Industrial Robot

Automatically controlled, reprogrammable multipurpose manipulator, programmable in three or more axes, which can be either fixed in place or mobile for use in industrial automation applications. The industrial robot includes the manipulator, including actuators, controller, including teach pendant, and any communication interface (hardware and software) [4].

Delays and latencies play an important role in robot control. They can occur at a different location in a robot control loop. These delays are influenced by initial, final, and processing speeds. Actuation delay and response delay are different types of delay in robotics. In this paper, we will try to find the best code and also will see if latency is affected by the speed of the robot.

2. METHODOLOGY

2.1 Experimental Setup and Procedure

The robot used in this experiment is ABB IRB 2400 as shown in figure 1. The IRB 2400 is a 6-axis industrial robot, designed specifically for manufacturing industries that use flexible robot-based automation. The robot is equipped with the IRC5 controller and robot control software, RobotWare. RobotWare supports every aspect of the robot system, such as motion control, development, and execution of application programs, communication, etc. Product specification - Controller IRC5 with FlexPendant. In figure 1, Pos A, B, C, D, E, and F represents Axis 1, 2, 3, 4, 5, and 6 respectively.

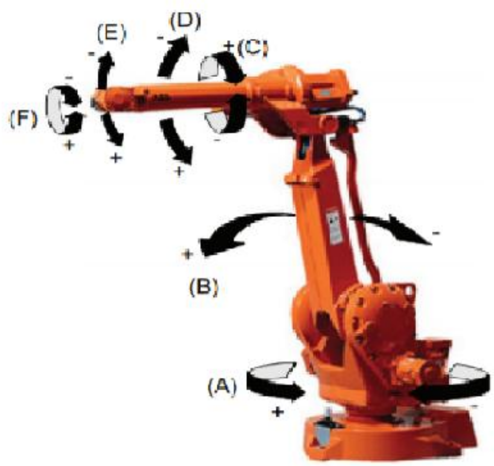


Figure 1: ABB IRB 2400 Robot [5].

It has been attached the inductive proximity sensor to the robot's arm and the workpiece consist of a rectangular metallic strip is placed on the table to experiment. The metallic strip is scanned by the sensor which is mounted on the robot. When the metal strip is detected, the current position is recorded, and it stops. The position where it stops is also noted. The difference between the two points-position of the strip and the position of the stop; gives us the latency of the robot. To find the latency of the robot we used two different codes. The 1st code uses the "search L" function whereas the other one used was "Trap routine".

Code for Search L Function

```
PROC main()
Open "HOME:" \File:= "POSITIONS.doc", logfile \Append;
MoveL p20, v150, fine, toolsenor;
SearchL\PStop, diSignalC1, sp, p10, v250, toolsenor;
writefile;
ENDPROC
PROC writefile()
```

```
signalposition:= sp.trans;
finalpos := cPos () ;
Write logfile, "signalposition="\pos:=signalposition;
Write logfile, "finalposition="\pos:=finalpos;
ENDPROC
```

Code for TRAP Routine

```
PROC main()
Open "HOME:" \File:= "POSITIONS.doc", logfile \Append;
MoveJ p20, v200, fine, toolsenor; CONNECT detect1 WITH measure;
 ISignalDI diSignalC1,1,detect1;
MoveL p10, v300, fine, toolsenor;
ENDPROC
PROC writefile()
signalposition:= sp.trans; finalpos := cPos () ;
Write logfile, "signalposition="\pos:=signalposition;
Write logfile, "finalposition="\pos:=finalpos;
ENDPROC
TRAP measure
sp:=CRobT(); StopMove;
writefile;
ENDTRAP
ENDMODULE
```

3. RESULTS AND DISCUSSIONS

From this experiment, by using the code with Search 'L' instruction, it stocks the position of the current robot when an input signal is noticed. During the run of the robot when this program was applied, we could see that the robot after stopping was moved a certain distance and did not go back to its previous position that is the searched position where there was a change in a signal variable.

Whereas while using the code with Trap Routine, the connecting interrupts are fascinating because they do give a chance to the user to make changes whenever the input signal gets detected. This code could be useful for initiating a short-term movement eventually making it restart the actual movement.

3.1 Analysis

There is a very minimal difference in latency by using separate codes. After applying the codes “SearchL” and “Trap Routine” on the robot, the values are given below, and the graph has been plotted for the speed of working and both distances. The graph shows us when the speed of working of the robot increases thereby increases latency which makes the robot stop for an extended distance. The SearchL shows less latency compared to Trap Routine as Trap Routine draws more time to stop than SearchL after the signal gets detected. From table 1, we know that the latency is low when the speed is at 10mm/sec while it is high when it is at 300mm/sec.

Table 1: Initial and Final Sensed Position at Different Speeds using Search L and Trap Function

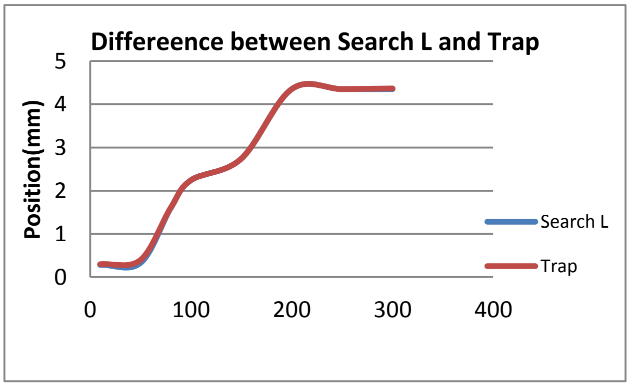
	Using Search L function		Using Trap function	
Speed	**Signal Position in X**	**Final Position in X**	**Signal Position in X**	**Final Position in X**
10	2020.88	2020.59	2021.1	2020.07
50	2020.98	2020.65	2021.06	2021.7
80	2020.91	2019.3	2021.05	2019.89
100	2020.9	2018.66	2021.18	2018.94
150	2020.98	2018.24	2021.17	2018.43
200	2021.03	2016.68	2021.17	2016.82
250	2021.03	2016.68	2021.17	2016.82
300	2021.03	2016.68	2021.17	2016.82

4. CONCLUSIONS

Two types of codes were used and compared to measure the latency in this experiment. The comparison can be seen in table 2 and graph 1. SearchL is more effective than Trap Routine. From the analysis, we can say that the latency is higher when Trap Routine was applied than SearchL. Also, the stopping distance was high compared to SearchL for few speeds. Future research in this line would be to find a more accurate way of measuring latency in the system. That will allow us to create a simpler functional code that focuses only on the task at hand.

Table 2: Difference between the Sensed and Stopping Point

Speed (mm/s)	Using search L Function (mm)	Using Trap Function (mm)
10	0.29	0.3
50	0.33	0.4
80	1.6	1.61
100	2.24	2.24
150	2.74	2.74
200	4.35	4.35
250	4.35	4.35
300	4.35	4.36



Graph 1: Difference between Search & Trap.

REFERENCES

1. *M. S. A. E. F. W. O. T. M. S. S. B. a. R. R. Alexander Gloye, "Predicting away robot control latency," in Robot Soccer World Cup, 2003*
2. *F. Enner, "A Practical Look at Latency in Robotics: The Importance of Metrics and Operating Systems," 2016.*
3. *"Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Robot.*
4. *"Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Robot.*
5. *[Online].Available: https://library.e.abb.com/public/6b65216f41634b838ad78d6d63b76b69/3HAC042195%20PS%20IRB%202400-en.pdf.*

.